Gmail rizwan atta <rizwanatta890@gmail.com>

Fwd: Alexa skill: Protogo

1 message

rizwan@infosys-sol.com <rizwan@infosys-sol.com> Wed, Mar 22, 2017 at 1:38 AM
To: rizwanatta890@gmail.com

——— Original Message ———
Subject: Fwd: Alexa skill: Protogo
Date: 2017-03-21 14:36
From: rizwan@infosys-sol.com
To: rizwanatta890@gmail.co

——— Original Message ———
Subject: Alexa skill: Protogo
Date: 2017-03-21 10:42
From: Taqi Jaffri <tjaffri@gmail.com>
To: rizwan@infosys-sol.com

Hi Rizwan,

Please start on the following alexa skill. Hopefully the steps below are clear but let me know if you have any questions.

GENERAL SETUP
1. What is your GitHub account? I need to give you access to the Git repo to commit this code. Do not commit any code to a free/public repo since the code is private to a client. I will give you access to a private repo and you can fork that for working. For now work locally and we can talk tonight about committing to a repo.
2. For now, create the AWS lambda and the Alexa skill in your own Amazon account. I will talk to you today or tomorrow to figure out how to switch to the client's official account. For now just write the code.
3. The API documentation is attached (two documents). This is the REST API provided by the client, which will be called internally by your skill to get data about procedures. Think of how the alexa-skill-meezan internally makes API calls to the meezan api that's what you will be doing here, i.e. your alexa skill will be calling. Please read and comment.

SUGGESTED PROCESS
1. Start the skill using the "yo alexa-skill" command. Specifically look at the yeoman starter here:
https://www.npmjs.com/package/generator-alexa-skill
2. Create some very basic test responses. I would suggest just a simple "help" or "about" response for now, just to get started and get the skill working end to end. Refer to
https://github.com/tango2bravo/alexa-skill-homr and
https://github.com/tjaffri/alexa-skill-meezan if you have any questions.
3. Make sure you create some very basic/trivial unit tests and update all the README and build scripts in package.json etc. Important to get all the infrastructure correct from the beginning.
4. For now, ignore the circle.yml or travis.yml files. I will help set up continuous integration later. For now for testing you can locally build the skill (see README files for my other skills linked above) and upload manually to AWS lambda to get things working.
5. Start thinking about how you will implement the full conversation flow. There are some tricky things, e.g. navigating between steps in the

procedures and remembering state for the user i.e. when the user says next you need to know where they were in order to go next. I can help, but if you read the meezan skill in detail it is already doing some of this using session object (that's what you will need to use as well).

SKILL SAMPLE CONVERSATION
We need to build a skill that can do the following conversation. We should talk in detail about this, but see how much you can read the API documents attached and start forming your own thoughts about how you would implement this.

ACTOR

ACTION

NOTES

User

Says: “Alexa, ask Protogo what is the startup procedure for a Naptha Hydrotreater”

“startup procedure for Naptha Hydroptreater” may be replaced by the name of any other procedure.

Alexa

Responds: “I’ve found the startup procedure for Naptha Hydrotreater. It’s purpose is ABC. It has 4 detail sections. One, Hazards. Two, Instructions… Which section would you like to hear?”

If there are multiple procedures matching the supplied query, the first one will be used.

For the matching procedure, the purpose and details about sections will be pulled from client-supplied REST API.

User

Responds: “Instructions”

The user responds to Alexa’s question from the previous query. If the user says something Alexa does not understand in this context, she will ask the user to repeat and will restate her previous question.

Alexa

Responds: “Here are the Instructions for Startup Procedure for Naptha Hydrotreater. One, Notify Supervisor. Two, ensure all pre-requisites have been met, Three…”.

Alexa starts reading a particular section.

User

Says: “next section”

The full set of pagination commands that will be supported are:

1. Next Section
2. Previous Section
3. Next Step

4. Previous Step

5. Pause

6. Resume

Alexa

Next section. Here are the Hazards for Startup Procedure for Naptha Hydrotreater. One, Fire. Two, Explosion. Three, Chemical. End of Section. Next section is Personal Protection Equipment. Would you like me to read it?

Alexa cues the next section when one section is over, asking the user if she should keep reading.

User

No, thanks

Session ends. The user can also end a session at any time by telling Alexa to stop (as opposed to pause which does not end a section)

Alexa

Goodbye

User

Alexa, ask Protogo to read Hazards for Startup Procedure for Naptha Hydrotreater.

Note that since the session has ended the skill now needs to be invoked with the full skill name (i.e. Alexa, ask Protogo to…). In this case the user specified a particular section of the procedure to start reading, as opposed to starting from the list of sections as in the first row.